Softwarica College of IT & E-Commerce
STW210CT: Programming, Algorithms and Data Structures
Assignment Brief 2021

Module Name STW210CT: Programming, Algorithms and Data Structures	Ind/Group **Individual**	Cohort **Feb 2021**	Module Code: **STW210CT**
Coursework Title: Exam replacement CW			Hand out date: TBD
Lecturer: Hikmat Saud			Due date: TBD
Estimated Time (hrs): Word Limit*: n/a	Coursework type: Individual / Practical		% of Module Mark 30%
Submission arrangement online via Softwarica Moodle: Upload through Assignment links File types and method of recording: URLs (source code repositories) Mark and Feedback date: Within 3 weeks of assignment submission Mark and Feedback method: Rubric marks and comments via Softwarica LMS			

Module Learning Outcomes Assessed:

1. Write software to solve a range of problems.
2. Implement and use simple searching and sorting algorithms.
3. Use libraries to extend the functionality of the base language.
4. Use basic design and testing strategies

Notes:

1. You are expected to use the CUHarvard referencing format. For support and advice on how this student can contact Centre for Academic Writing (CAW).
2. Please notify your registry course support team and module leader for disability support.
3. The University cannot take responsibility for any coursework lost or corrupted on disks, laptops, or personal computer. Students should therefore regularly back-up any work and are advised to save it on the University system.
4. If there are technical or performance issues that prevent students submitting coursework through the online coursework submission system on the day of a coursework deadline, an appropriate extension to the coursework submission deadline will be agreed. This extension will normally be 24 hours or the next working day if the deadline falls on a Friday or over the weekend period. This will be communicated via email and as a Softwarica Moodle announcement.

Question No 1

You are assigned a software development project to manage bookstore. To manage overall data related to books organization need bookstore inventory management system. Taking this scenario, develop a new computer-based system fulfilling the below requirements: -

1. Login system for Admin.
2. Add/update/delete book details
3. Record of available and sold books
4. Admin need to search book details according to book name, publisher name and publish date. For search implement appropriate searching algorithm with appropriate sorting mechanism to display information in ascending as well as descending order based on user input.

Implement the unit testing frameworks like JUnit to perform unit testing for at least 3 functionalities within this application. Students are encouraged to use GUI frameworks like Swing while building the application.

[Total 30 marks]

Marking Notes

1. All submitted coursework will be assessed via VIVA conducted at the end of this semester.
2. Each VIVA will last 20 minutes.
3. You will submit on the deadline a document (PDF or Word) on Moodle containing all the coursework tasks solved and including a link to your GitHub Classroom repository shared via Softwarica LMS.
4. During the VIVA you will be assessed with few relevant random questions.
5. If you submit only some of the tasks, your mark will be proportional to that.
6. The marking criteria valid for week 8-11 is presented below.

Criteria	0	1	2	3	4	5
Feature complete (5)	Not submitted	Only few features implemented and are not executing	Many of the features are implemented but are not executing correctly	Many of the features are implemented and are executing correctly	Most of the features are implemented and are executing correctly	All features implemented and are executing correctly
Code aesthetic (5)	Not submitted	Assignment submitted but not commented and formatted. variable's/classes/ function are defined but meaningless	Lack of comments, formatted in Source code. Only few classes and functions are defined but hard to read	Lack of comments, formatted in Source code, but meaningful variable/class/ function names are used few functions are defined.	Lack of comments, formatted in Source code, but meaningful variable/class/ function names are used. Code is easy to read	Source code is well commented, properly formatted, meaningful variable/function/class names are used. Code is easy read and understand, having many pure functions.
GUI (5)	Not submitted	Hard to use. Only some components are used and unmanaged	Few frames are difficult to use. UI component are used but unmanaged.	Some frames are difficult to use. UI component are used but unmanaged.	Easy to use, Proper use of various UI components. User Interaction is low	Easy to use, Proper use of various UI components, Clean and interactive UI
I/P Validation (5)	Not submitted	Only few input fields are validated. Error message are not shown	Only few inputs field are validated. Error messages are shown in code format	Most input fields are properly validated. Error messages are shown in code format	Most input fields are properly validated. Error messages are properly shown in natural language	All input fields are properly validated. Error messages are properly shown in natural language.
Unit Testing (5)	Not submitted	Only few features are tested without using framework and many of them are fail	Many of the modules are tested and many of them are fail	Many of the modules are tested using suitable unit testing framework.	Most of the modules are tested using suitable unit testing framework. Should have partial coverage.	All modules are unit tested using suitable unit testing framework. Should have full testing coverage.
Viva (5)	Not present	Could not explain	Could explain basic	Could explain	Could explain	Could explain

	(Assignment submitted but absent in viva)	reasoning behind the code. But answered only one viva question	terms but not about algorithm. But answered only two viva question	reasoning behind the code, including use of loops, conditions, algorithms. answered only three viva question	reasoning behind the code, including use of loops, conditions, algorithms. answered only four viva question	reasoning behind the code, including use of loops, conditions, algorithms. Answered all five question